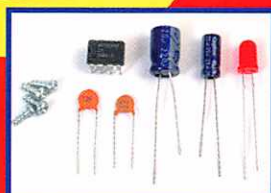
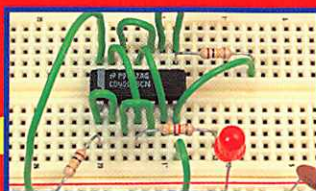
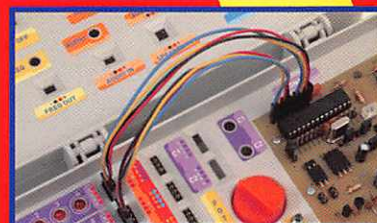
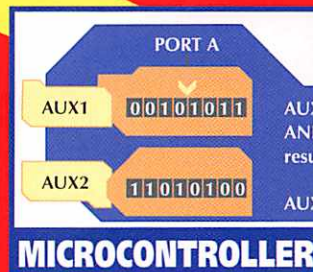
...e costruisci il tuo LABORATORIO DIGITALE

6,90 €

54

HARDWARE

DIGITALE DI BASE

MICROCONTROLLER

DIGITALE AVANZATO

TOTALMENTE PROGRAMMABILE!!!

9 771824 363008 41054

Peruzzo & C.

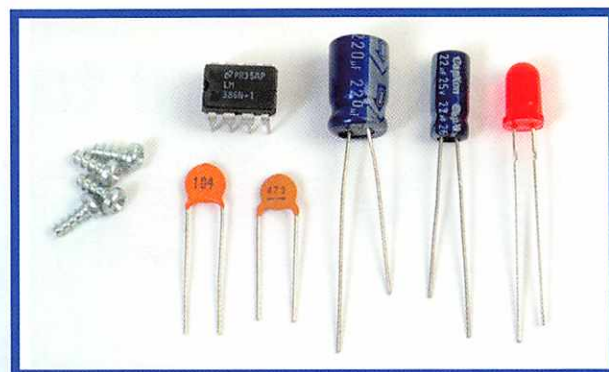
Direttore responsabile:
ALBERTO PERUZZO
Direttore Grandi Opere:
GIORGIO VERCELLINI
Consulenza tecnica e traduzioni:
CONSULCOMP S.n.c.
Pianificazione tecnica
LEONARDO PITTON

Direzione, Redazione, Amministrazione: viale Ercole Marelli 165, Tel. 02/242021, 20099 Sesto San Giovanni (MI). Pubblicazione settimanale. Registrazione del Tribunale di Monza n. 1738 del 26/05/2004. Spedizione in abbonamento postale gr. II/70; autorizzazione delle Poste di Milano n. 163464 del 13/2/1963. Stampa: Grafiche Porpora s.r.l., Cernusco S/N (MI). Distribuzione SO.DI.P. S.p.A., Cinisello Balsamo (MI).

"ELETTRONICA DIGITALE"
si compone di
70 fascicoli settimanali
da suddividere
in 2 raccoglitori.

RICHIESTA DI NUMERI ARRETRATI. Per ulteriori informazioni, telefonare dal lunedì al venerdì ore 9.30-12.30 all'ufficio arretrati tel. 02/242021. Se vi mancano dei fascicoli o dei raccoglitori per completare l'opera, e non li trovate presso il vostro edicolante, potrete riceverli a domicilio rivolgendovi direttamente alla casa editrice. Basterà compilare e spedire un bollettino di conto corrente postale a PERUZZO & C. s.r.l., Ufficio Arretrati, viale Marelli 165, 20099 Sesto San Giovanni (MI). Il nostro numero di c/c postale è 42980201. L'importo da versare sarà pari al prezzo dei fascicoli o dei raccoglitori richiesti, più le spese di spedizione € 3,10 per pacco. Qualora il numero dei fascicoli o dei raccoglitori sia tale da superare il prezzo globale di € 25,82 e non superiore a € 51,65, l'invio avverrà per pacco assicurato e le spese di spedizione ammonteranno a € 6,20. La spesa sarà di € 9,81 da € 51,65 a € 103,29; di € 12,39 da € 103,29 a € 154,94; di € 14,98 da € 154,94 a € 206,58; di € 16,53 da € 206,58 in su. Attenzione: ai fascicoli arretrati, trascorse dodici settimane dalla loro distribuzione in edicola, viene applicato un sovrapprezzo di € 0,52, che andrà pertanto aggiunto all'importo da pagare. Non vengono effettuate spedizioni contrassegno. Gli arretrati di fascicoli e raccoglitori saranno disponibili per un anno dal completamento dell'opera. IMPORTANTE: è assolutamente necessario specificare sul bollettino di c/c postale, nello spazio riservato alla causale del versamento, il titolo dell'opera nonché il numero dei fascicoli e dei raccoglitori che volete ricevere.

IN REGALO in questo fascicolo

1 Integrato LM386
1 Condensatore elettrolitico 220 µF
1 Condensatore elettrolitico 22 µF
1 Condensatore ceramico 47 nF
1 Condensatore ceramico 100 nF
1 LED rosso
4 Viti

IN REGALO nel prossimo fascicolo

1 Altoparlante da 8 Ω

COME RACCOGLIERE E SUDDIVIDERE L'OPERA NELLE 4 SEZIONI

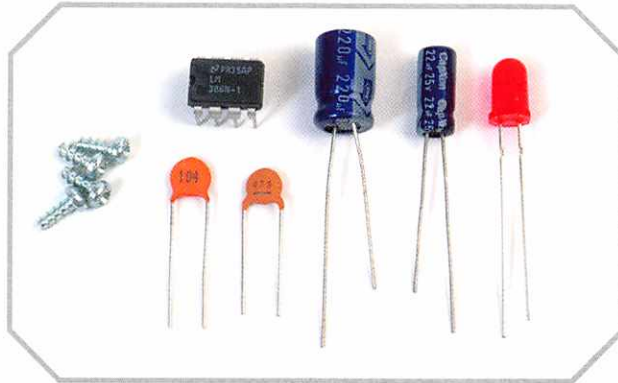
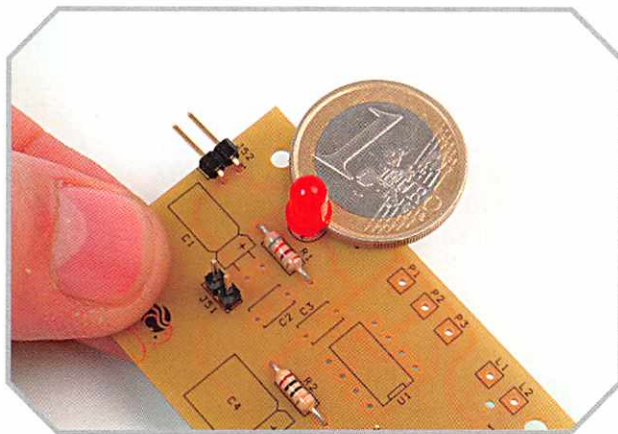
L'Opera è composta da 4 sezioni identificabili dalle fasce colorate, come indicato sotto. Le schede di ciascun fascicolo andranno suddivise nelle sezioni indicate e raccolte nell'apposito raccoglitore, che troverai presto in edicola. Per il momento, ti consigliamo di suddividere le sezioni in altrettante cartellette, in attesa di poterle collocare nel raccoglitore. A prima vista, alcuni numeri di pagina ti potranno sembrare ripetuti o sbagliati. Non è così: ciascuno fa parte di sezioni differenti e rispecchia l'ordine secondo cui raccogliere le schede. **Per eventuali domande di tipo tecnico scrivere al seguente indirizzo e-mail:** elettronicadigitale@microrobots.it

Hardware Montaggio e prove del laboratorio

Digitale di base Esercizi con i circuiti digitali

Digitale avanzato Esercizi con i circuiti sequenziali

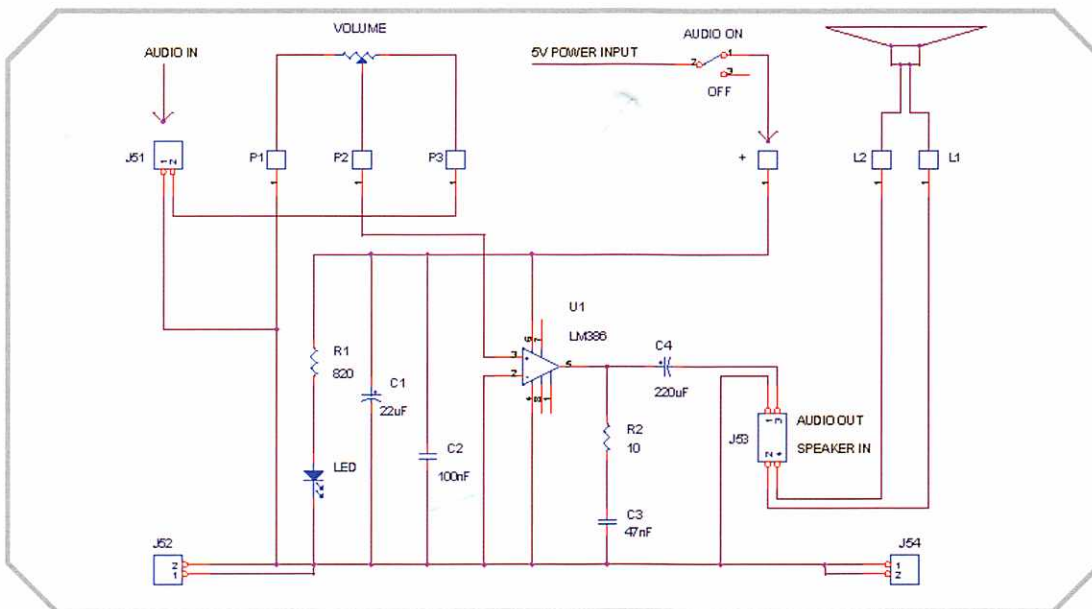
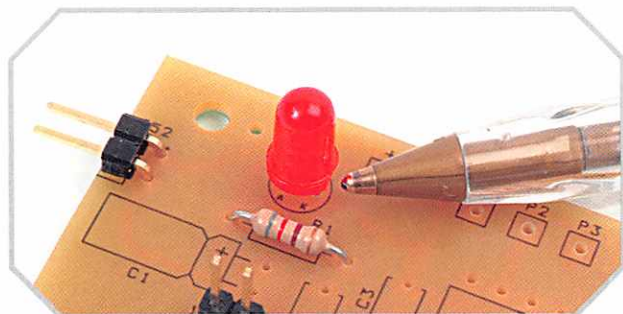
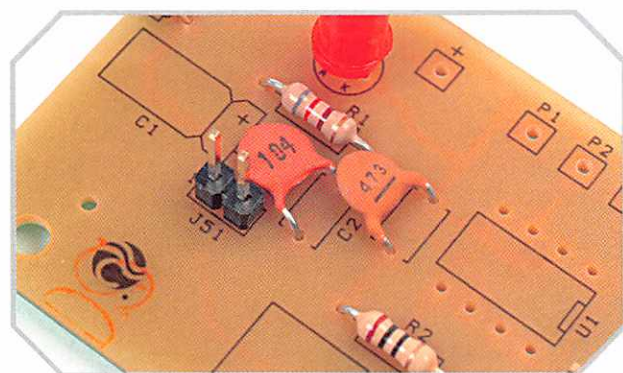
Microcontroller Esercizi con i microcontroller

L'amplificatore audio (II)

Componenti allegati a questo fascicolo.

Il LED deve rimanere separato dal circuito.

Con questo fascicolo vengono forniti i componenti necessari per terminare il montaggio della scheda DG15, corrispondente all'amplificatore audio, con le quattro viti necessarie per il suo montaggio.

Questo amplificatore audio basato sul circuito integrato LM386, ha un guadagno di tensione di 20 e può fornire una potenza di uscita di circa 1/4 di W, su un carico da 8 Ω; la tensione massima applicabile al suo ingresso è di +/- 4 mV. Se la tensione da applicare al suo ingresso è più elevata, dovrà essere ridotta con un partitore di tensione, o utilizzare a questo scopo il potenziometro "VOLUME". Può essere utilizzata con l'altoparlante che monteremo sul laboratorio oppure con uno esterno.

Il LED

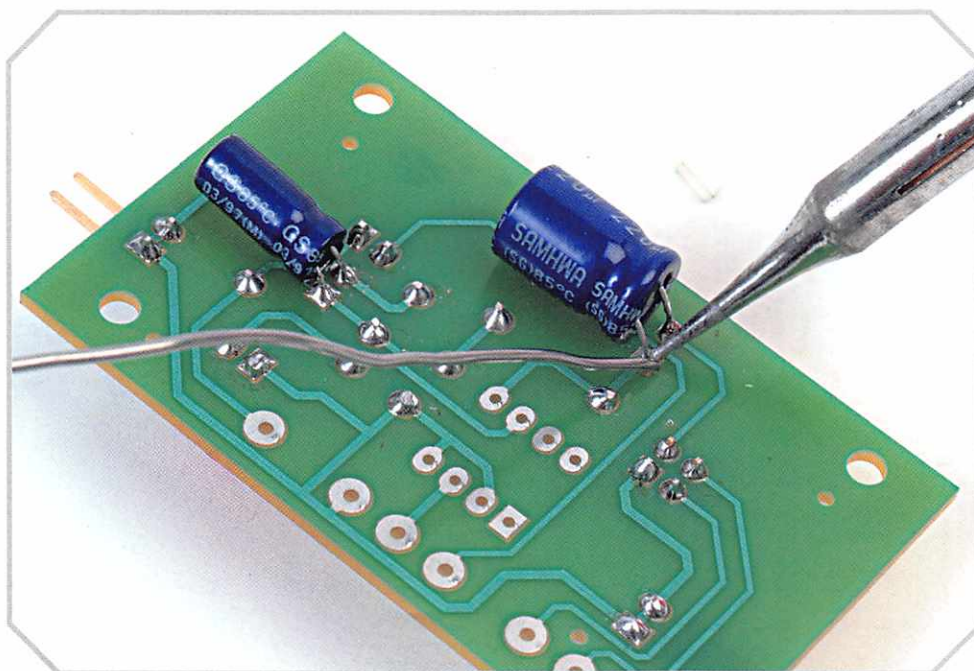
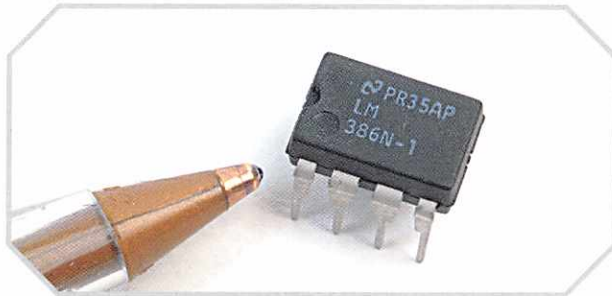
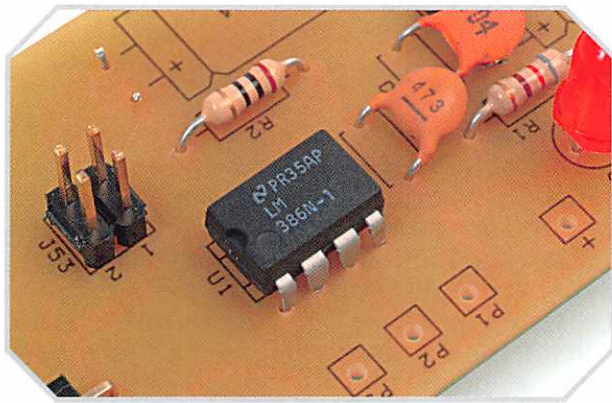
Questo circuito dispone di un diodo LED che si illumina quando riceve alimentazione, la sua installazione è semplice, anche se bisogna tenere presente che dispone di polarità, quindi è necessario sapere che il terminale corrispondente al catodo – terminale negativo – ha una zona piatta sulla base del contenitore. Il LED deve rimanere separato dalla scheda di circa

Schema elettrico dell'amplificatore audio e suoi collegamenti.

Il catodo del LED è identificabile grazie a una parte piatta sul suo contenitore.

I condensatori ceramici devono rimanere piegati.

2 mm, anche se questa misura non deve essere necessariamente così precisa, è sufficiente utilizzare una moneta da 1 euro o 50 centesimi o una simile, per ottenere questa separazione. Dobbiamo tenere fermo il LED mentre saldiamo i suoi terminali, in modo che rimanga diritto, però la cosa più importante è rispettarne la polarità.

Condensatori ceramici

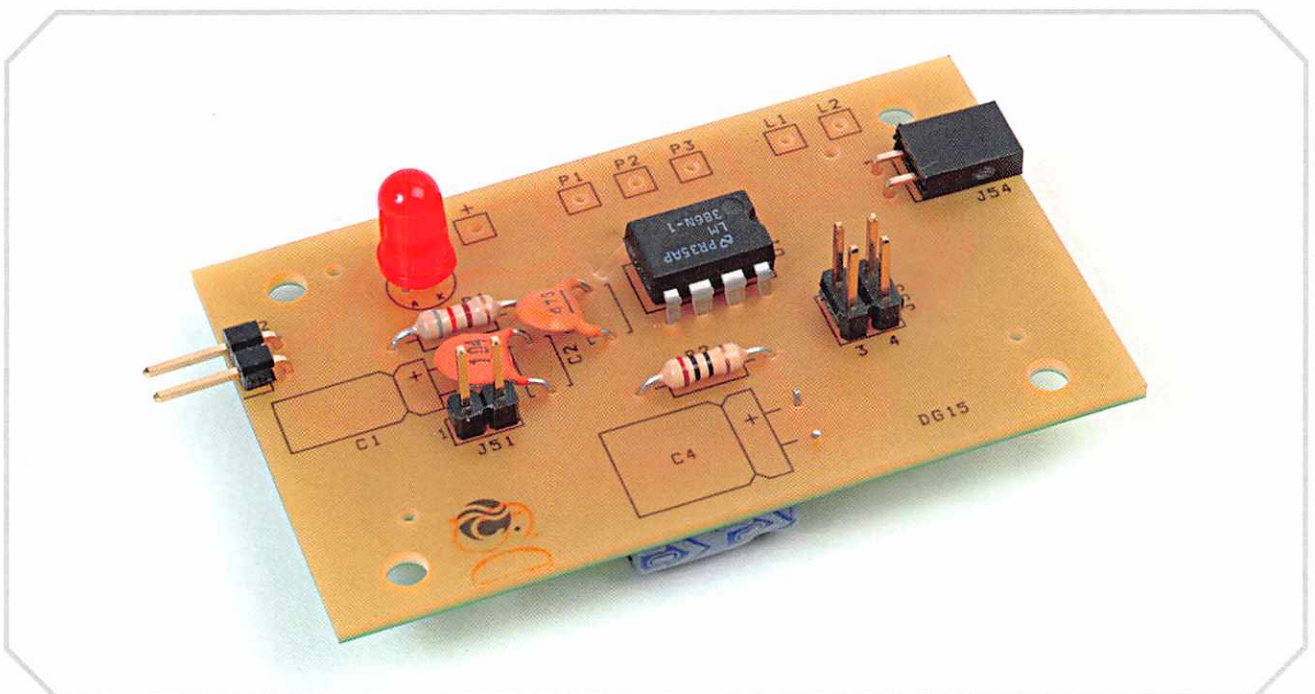
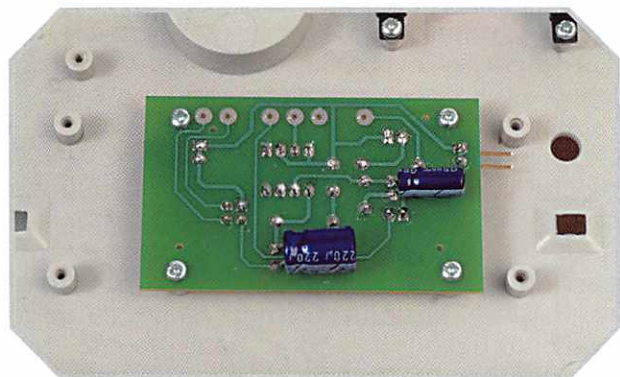
I condensatori ceramici si identificano come C2 da 100 nF e C3 da 47 nF. Si inseriscono come d'abitudine dal lato dei componenti del circuito stampato e non hanno polarità. Li dobbiamo piegare per ridurre l'altezza della scheda, in modo che dopo averli saldati, rimangano nella posizione che si può osservare nelle fotografie. Dobbiamo piegare i terminali, montare i condensatori nella loro posizione definitiva e fissarli. Poi li salderemo e li terremo fermi fino a quando la saldatura non si indurirà, in questo modo eviteremo che si muovano durante il processo di saldatura.

Condensatori elettrolitici

Questo circuito stampato contiene due condensatori elettrolitici, i cui riferimenti sono C1 da 22 μF e C4 da 220 μF. È molto importante ricordare che questi elementi hanno polarità,

I condensatori elettrolitici vengono saldati dal lato delle saldature.

Questo è il terminale 1 dell'integrato LM386.

Montaggio corretto del LM386.

cioè un terminale positivo e un terminale negativo. Sulla scheda è riportato il terminale positivo e sul contenitore del componente generalmente è indicato quello negativo. In ogni caso ogni terminale deve essere inserito nel foro che corrisponde alla sua polarità. Questa scheda ha una limitazione di altezza dal lato dei componenti, quindi è necessario montare i condensatori elettrolitici dal lato delle saldature. Per facilitare questo montaggio vengono posizionati piegati, però prima di piegare i terminali è necessario considerare con attenzione, quale sia il terminale positivo e quale il negativo. Dopo aver inserito i terminali si verifica nuovamente la polarità, si salda, e successivamente si taglia la parte in eccesso dei reofori.

LM386

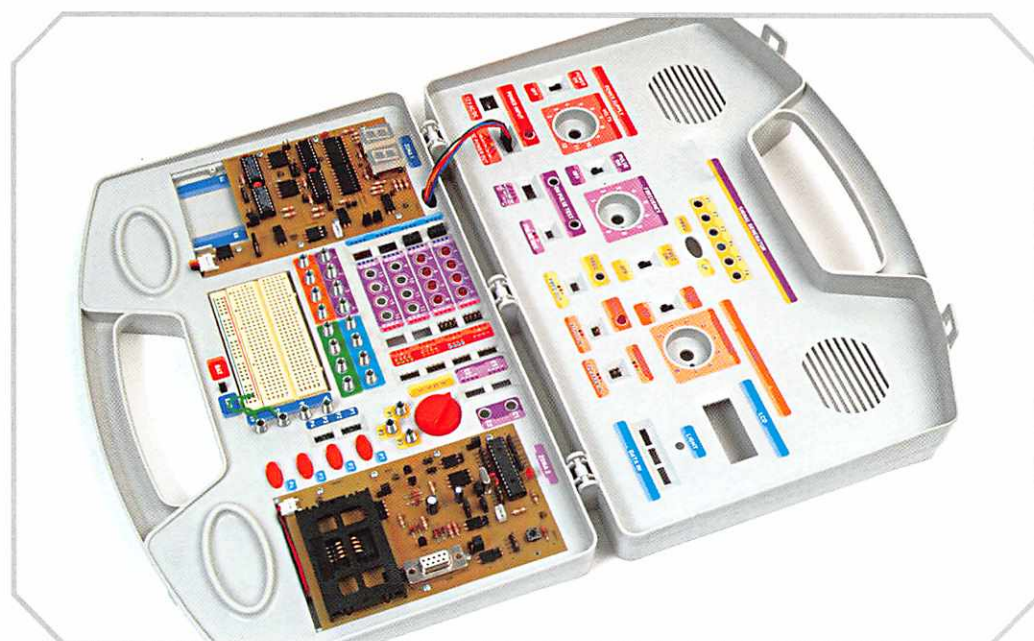
Questo circuito integrato viene fornito con un contenitore DIL-8, e ha un formato molto comune fra i circuiti integrati. È molto importante saper distinguere il suo terminale 1, per poterlo inserire correttamente sul circuito stampato. La tacca di riferimento può essere un punto sul terminale 1 o un'incisione fra il terminale 1 e il terminale 8. Questo è fondamentale, dato che l'integrato funziona solamente

Scheda DG15 completa.

Scheda DG15 avvitata.

Aspetto dei LED e dei connettori di DG15.

in questa posizione, inoltre se viene collegato male è molto probabile che lo si distrugga.

Per saldare questo circuito integrato dopo aver verificato la sua posizione, fisseremo inizialmente due dei suoi piedini in diagonale e controlleremo nuovamente il verso di montaggio; se risulta corretto continueremo con le saldature dei restanti terminali. Vi consigliamo di saldare i terminali in modo alterno, applicando il saldatore solo per il tempo necessario e applicando stagno di buona qualità, inoltre dobbiamo attendere qualche secondo fra una saldatura e l'altra, per evitare di surriscaldare il circuito integrato.

Montaggio della scheda DG15

Dopo aver verificato seguendo lo schema che la scheda DG15 abbia tutti i componenti montati, è possibile installarla sul pannello superiore del laboratorio. I LED dei connettori J51 e J52 devono fuoriuscire dal pannello principale, i connettori devono rimanere perfettamente centrati e la scheda appoggiata sulle quattro torrette, su cui monteremo le quattro viti, ma senza stringerle fino in fondo, in modo da permettere un minimo di movimento laterale della scheda. Inseriremo un ponticello sul connettore J51 e sul connettore J53. Ora è possibile avvitare le viti, ma senza stringerle eccessivamente, e con la presenza dei ponticelli inseriti che a questo punto possono essere estratti, vi è una maggior garanzia che i connettori rimangano posizionati al centro dei fori.

Vista generale del laboratorio.

Bistabile con priorità

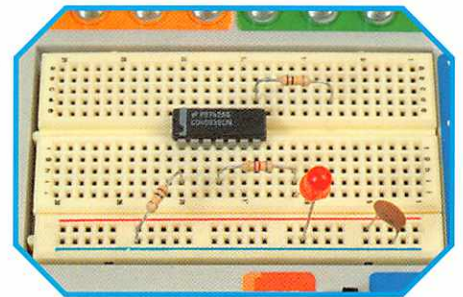
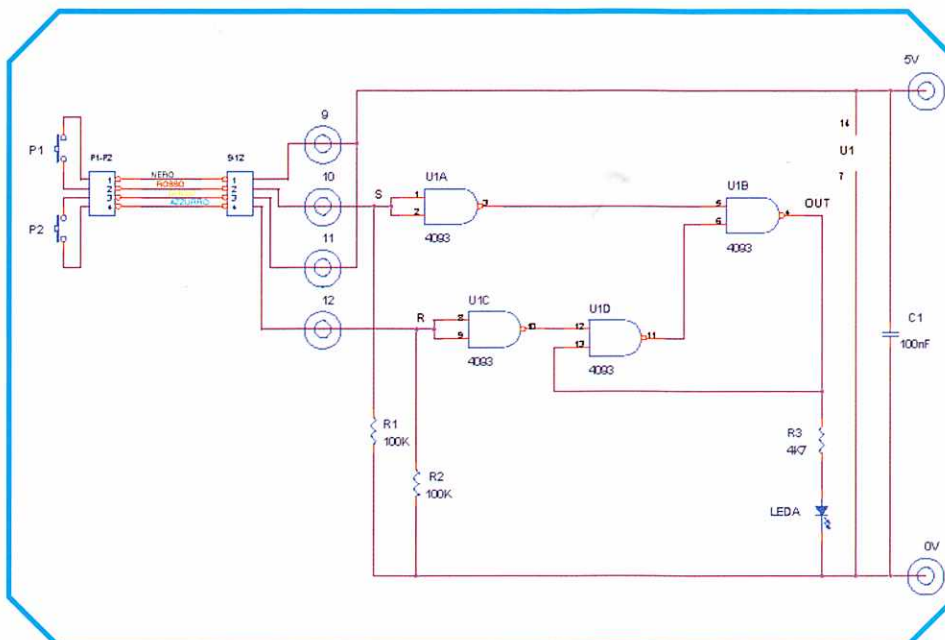
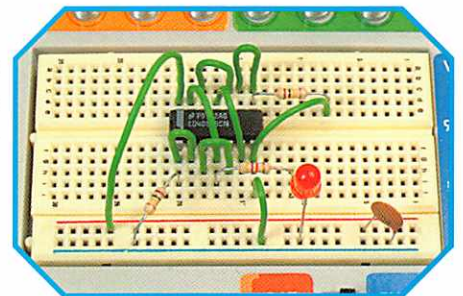
Se ricordiamo la tabella della verità di un bistabile con porte NAND, quando applichiamo su entrambi gli ingressi e contemporaneamente un 1 logico, l'uscita non cambia e si mantiene nello stato precedente.

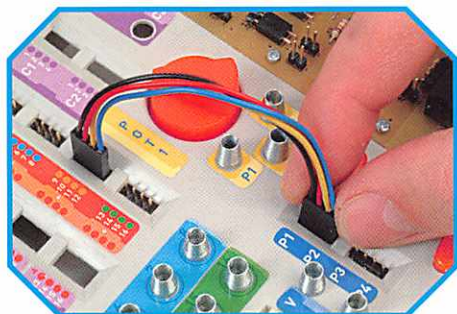
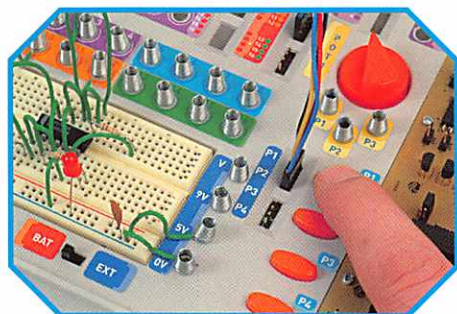
In questo nuovo circuito invece la combinazione uno-uno sull'ingresso origina un uno sull'uscita, mentre i terminali SET e RESET mantengono la loro funzionalità. La combinazione zero-zero sull'ingresso non si deve utilizzare, dato che non garantisce alcun risultato prevedibile sull'uscita.

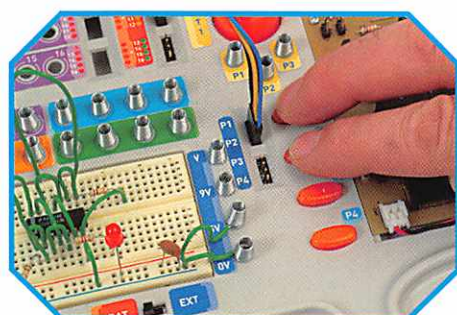
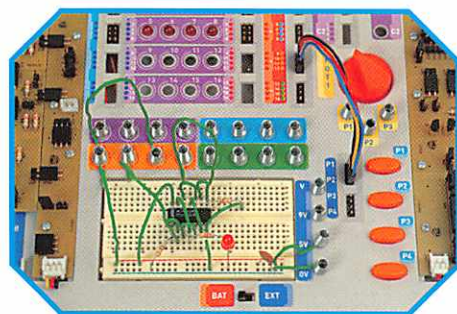
Il circuito

Osservando lo schema vedremo che si utilizzano le quattro porte del circuito integrato, la cosa più interessante da notare comunque è il terminale 4 dell'uscita collegato sul terminale 13 di un'altra porta. Quando nessun pulsante è premuto gli ingressi sono a 0, grazie all'utilizzo delle resistenze di pull-down R1 e R2, e l'uscita in teoria potrebbe assumere qualsiasi valore.

Se si aziona il pulsante P1 si applica un 1 sul-

Componenti sulla scheda Bread Board.

Collegamenti della scheda Bread Board.

Schema del bistabile con priorità a 1.

LISTA DEI COMPONENTI	
U1	Circuito integrato 4093
R1, R2	Resistenza 100 K (marrone, nero, giallo)
R3	Resistenza 4K7 (giallo, viola, rosso)
C1	Condensatore 100 nF
LED A	Diodo LED rosso 5 mm

Collegamenti al resto del laboratorio.

Il pulsante P1 è quello di SET.

Il pulsante P2 è quello di RESET.

Premendo contemporaneamente P1 e P2 l'uscita è 1.

Esperimento completato.

l'ingresso di SET, terminale 1 e 2 della porta U1A, mentre se si preme P2 si applica un 1 sulla porta U1C, terminali 8 e 9 dell'integrato 4093, e l'uscita passa a valore 1.

Tabella della verità

La tabella della verità di questo circuito è molto simile a quella di un bistabile, con la differenza che alla posizione 1-1 corrisponde l'uscita 1.

P1(S)	P2(R)	OUT
1	0	1
0	1	0
1	1	1
0	0	X

Questa tabella la potremo verificare dopo aver montato il circuito. Il pulsante P1 è quello di SET, e premendolo otterremo un 1 sull'uscita. Il pulsante P2 è quello di RESET e premendolo otterremo uno 0 sull'uscita. Premendo P1 e P2 l'uscita sarà 1.

Montaggio

Il montaggio è abbastanza semplice, però deve essere eseguito con attenzione per non commettere errori.

Molto importanti sono la posizione del circuito integrato e la polarità del LED, senza dimenticare di collegare l'alimentazione del circuito integrato, terminale 14 al positivo, e il terminale 7 al negativo. L'alimentazione può essere da 5 o da 9 V e il condensatore C1 è un filtro di alimentazione. I pulsanti P1 e P2 si collegano con un cavetto terminato su due connettori a quattro vie, utilizzando le molle 9 e 12.

L'esperimento

Dopo aver alimentato il circuito si utilizzano i pulsanti per verificare la tabella della verità, tenendo presente che se si preme si genera un 1 altrimenti c'è uno 0.

L'uscita 0 è indicata con il LED spento, mentre l'uscita 1 è segnalata dall'illuminazione del LED.

Esercizio 16: l'importanza dello SLEEP e del WDT, la pratica

Questo esercizio è un esempio di come si lavora con l'istruzione SLEEP combinata con il WDT. È molto importante saper gestire questa istruzione, perché saranno molte le applicazioni che progetteremo in cui il PIC dovrà rimanere in modo riposo o di basso consumo.

Scrittura del programma

Configurazione hardware del laboratorio
Per trasferire al PIC il codice in linguaggio macchina (ese16.hex) dobbiamo configurare il laboratorio. Collegheremo il cavo di comunicazione fra quest'ultimo e il PC, verificheremo il corretto posizionamento del PIC sul suo zoccolo e configureremo i ponticelli per questo modo lavoro.

Inseriremo i ponticelli sui connettori JP8 e JP9 e cambieremo di posizione i ponticelli sui connettori JP1, JP2 e JP3 portandoli sulle posizioni 1 e 2.

Software IC-Prog

Faremo partire IC-Prog e come vediamo dalla tabella nella figura, la prima cosa da fare è leggere il dispositivo nel caso contenesse qualche programma che meriti di essere conservato. Cancelleremo il contenuto del PIC e verificheremo questa fase per accertarci che sia avvenuta correttamente. Apriremo il nostro file e configureremo l'oscillatore, la protezione del codice e i bit della parola di configurazione.

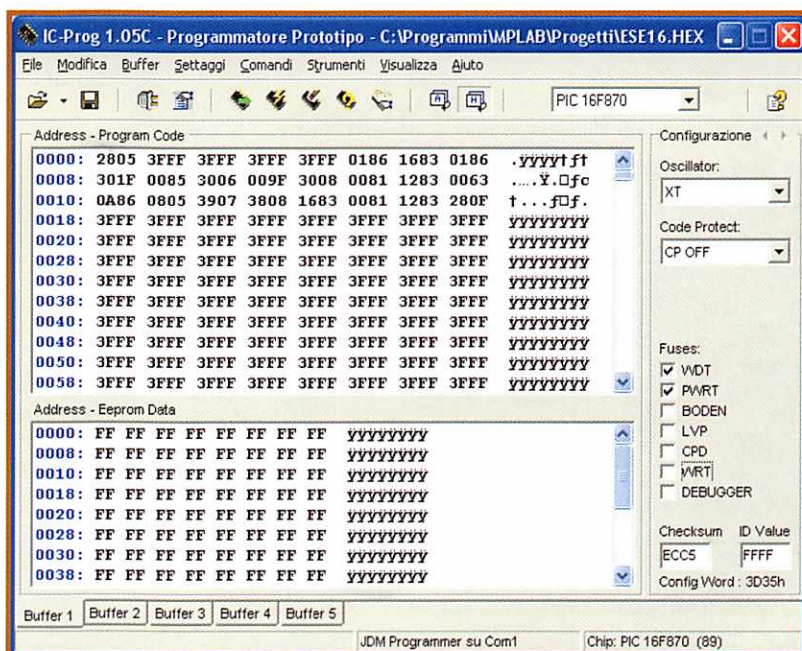
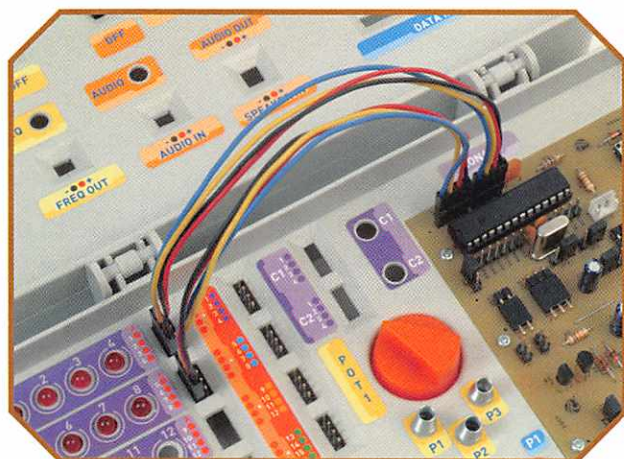
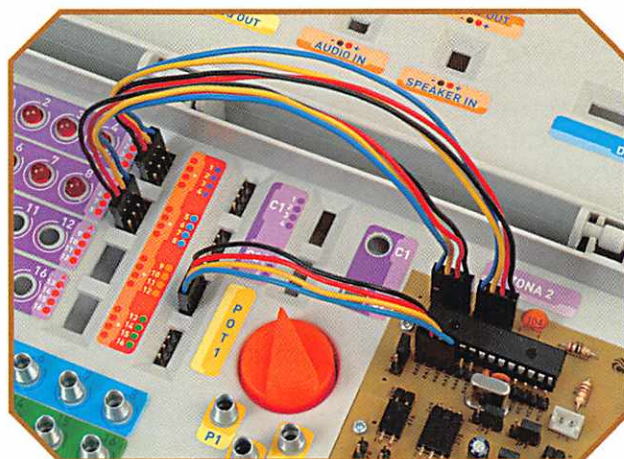
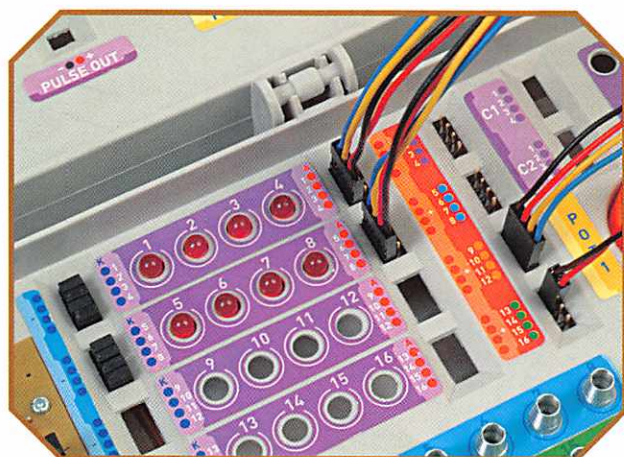
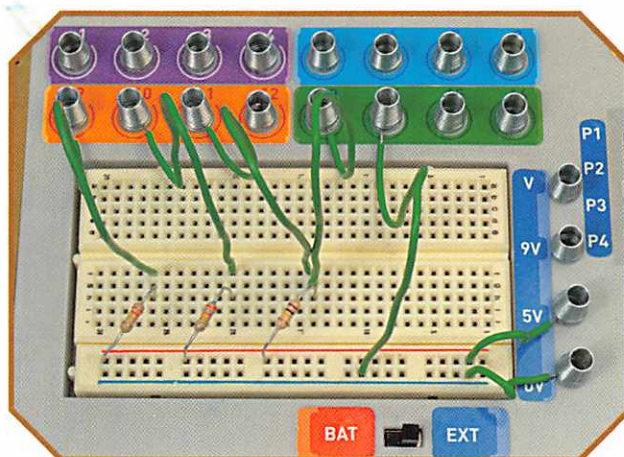
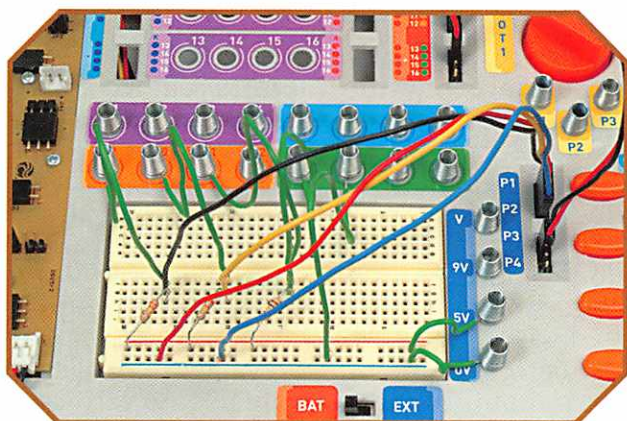
IC-Prog dopo la configurazione avrà l'a-

PASSAGGI PER SCRIVERE IL PIC	
1	Verificare il corretto inserimento del PIC
2	Collegare il cavo di comunicazione Laboratorio-PC
3	Inserire i ponticelli su JP8 e JP9
4	Ponticelli su JP1, JP2 e JP3 sulle posizioni 1 e 2
5	IC-Prog: Fase di lettura, cancellazione e verifica
6	IC-Prog: Carico del programma e configurazione del software
7	IC-Prog: Scrittura e verifica

Passaggi per scrivere un programma nel PIC.

Porta B		LED n°
RB0	→	4
RB1	→	3
RB2	→	2
RB3	→	1
RB4	→	8
RB5	→	7
RB6	→	6
RB7	→	5

Corrispondenza tra i terminali della porta B e i LED.

Ora possiamo selezionare l'opzione: Programma Tutto.

Inseriamo i ponticelli sui catodi dei LED.

Uniamo con un cavetto la porta B e gli anodi dei LED.

Uniamo la porta A con le molle di collegamento.

Utilizziamo le molle 13 e 14 per inserire il pulsante P3.

Colleghiamo il pulsante P3 sulla scheda Bread Board.

Colleghiamo gli altri due pulsanti direttamente sulla scheda.

spetto che possiamo vedere nella figura della prima pagina e sarà pronto per trasferire il programma al PIC. Selezioneremo l'opzione Programma Tutto e inizieremo a trasferire il programma, tenendo presente che al termine del trasferimento dovremo verificare che sia avvenuto con successo. Per eseguire questa verifica è sufficiente leggere il dispositivo e controllare che il programma che si trova scritto all'interno corrisponda a quanto stiamo cercando di scrivere.

Montaggio

Prima di eseguire il montaggio dell'applicazione dobbiamo riportare la configurazione del laboratorio per il modo lavoro, cioè toglieremo i ponticelli JP8 e JP9 e sposteremo JP1, JP2 e JP3, inoltre scollegheremo il cavo di trasferimento fra il PC e il laboratorio.

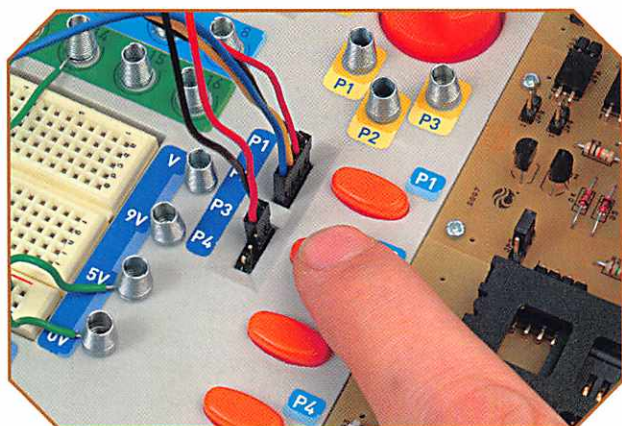
Montaggio delle uscite

Poiché ogni volta che il PIC si risveglia dal modo lavoro di basso consumo si incrementerà di una unità un contatore e dato che quest'ultimo verrà visualizzato sulla matrice dei diodi LED, dobbiamo collegare la porta B di uscita con la matrice. Prima di eseguire questo collegamento, verificheremo di aver collocato correttamente i ponticelli sui catodi dei diodi. Utilizzeremo due file della matrice, perché il contatore sarà da 8 bit, come si può vedere nella figura.

Mediante i cavetti uniremo la porta B con gli anodi dei diodi, tenendo presente che il contatore avrà il bit meno significativo nella fila superiore a destra e il bit più significativo nella fila superiore a sinistra. Nel grafico è possibile vedere la corrispondenza fra i terminali della porta B e i diodi LED della matrice, nelle immagini è possibile vedere i collegamenti.

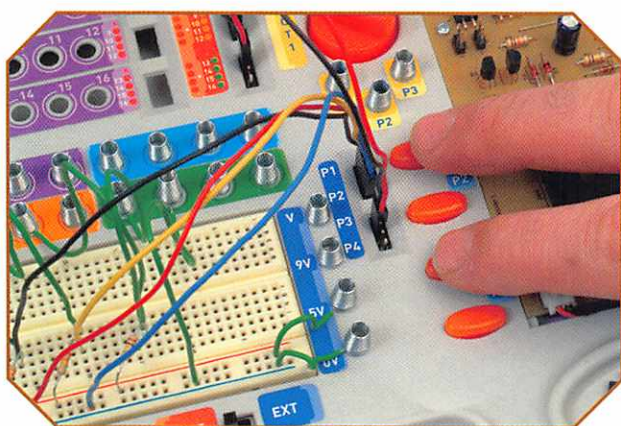
Ingressi

Mediante gli ingressi assegneremo il predivisore del Watchdog Timer, quindi avremo bisogno di tre segnali di ingresso, i quali arriveranno ai terminali RA2, RA1 e RA0. Tramite un cavetto uniremo i terminali della porta A e il connettore maschio corrispondente alla fila

Aspetto del laboratorio dopo aver concluso il montaggio.

Manteniamo premuto il pulsante per modificare il predivisore.

delle molle di collegamento colore arancio (molle dalla 9 alla 12). Con questo collegamento possiamo quindi eseguire il montaggio degli ingressi sulla scheda Bread Board.

Per simulare gli interruttori utilizzeremo i pulsanti P1, P2 e P3. Negli esercizi precedenti abbiamo visto come ottenere un 1 logico collegando al terminale di ingresso (o dalla sua molla di collegamento equivalente) una resistenza al positivo, per ottenere uno 0 invece uniremo il terminale di ingresso a massa (negativo o 0 V) mediante un cavetto. Per utilizzare i pulsanti sarà sufficiente inserirli fra il terminale di ingresso e la massa, in modo che quando vengono premuti svolgano la stessa funzione del filo che abbiamo utilizzato negli esercizi precedenti.

Monteremo tre resistenze sul positivo nella scheda Bread Board e l'altro capo lo uniremo alle molle di collegamento corrispondenti agli ingressi.

Proviamo diverse combinazioni.

Per collegare il primo dei pulsanti mediante un cavetto aggiuntivo collegheremo i due terminali del connettore maschio relativo al pulsante P3 con il corrispondente connettore delle molle di collegamento 13 e 14 (fila verde).

Una delle molle, la 13 ad esempio, la collegheremo al terminale di ingresso RA2 (molla 11) e l'altra al negativo. In questo modo quando premeremo il pulsante chiuderemo il suo contatto, e porteremo il negativo sull'ingresso del PIC.

Questo è un montaggio valido per collegare i pulsanti, ma come avete potuto verificare abbiamo dovuto utilizzare due molle in più di collegamento. I pulsanti si possono collegare direttamente alla scheda Bread Board senza passare attraverso le molle di collegamento, quindi lo faremo per i due ingressi rimanenti. Collegheremo uno dei capi del pulsante all'ingresso (alla sua molla corrispondente o al collegamento sulla scheda) e l'altro direttamente al negativo. Faremo lo stesso sia per P1 che per P2, in modo che il montaggio risulti simile a quello dell'immagine.

Prova di funzionamento

Abbiamo terminato il montaggio dell'applicazione e il laboratorio ha un aspetto come quello riportato nell'immagine della pagina precedente. Inizialmente gli ingressi sono tutti al positivo, quindi su RA2, RA1 e RA0 sta entrando '111', ciò significa che il predivisore acquisisce il valore di 128. Questo è il valore massimo che può assumere il predivisore del WDT e significa che la frequenza di incremento del contatore sarà la più bassa, circa 2,3 secondi.

Possiamo cambiare il range del predivisore solamente inserendo uno degli ingressi al negativo. Manterremo P2 premuto, ad esempio, e vedremo che la frequenza del contatore aumenta. Stiamo contando più rapidamente perché ora il segnale dell'ingresso è '101' quindi il predivisore assume un valore di 32 (quattro volte minore di quello iniziale).

Premendo P1 e P3 il range del predivisore è 4 e la frequenza di incremento del contatore è molto più elevata (ogni 72 ms).

Non dimenticate che il microcontroller passa la maggior parte del tempo in modo di basso consumo, uscendo da questo stato solo per incrementare il valore dell'uscita che indica quante volte si è "risvegliato".

Esercizio 15: la USART, il programma

***L**'ultimo degli esercizi previsti per consolidare le conoscenze della teoria, tratta la comunicazione seriale con il modulo USART.*
Questo dispositivo ci permette di comunicare con altri dispositivi, ed è molto utilizzato per far comunicare il PIC con un PC.

Comunicazione PIC-PC

In molte applicazioni si utilizza un PC per eseguire un controllo o la supervisione dell'applicazione, però è il PIC che ha il compito di agire sul sistema. Per poter mettere in comunicazione il PIC con il PC utilizzeremo la comunicazione seriale RS 232 e a questo scopo il PIC deve utilizzare il suo modulo di comunicazione seriale USART nel modo asincrono. Stabilita la comunicazione potremo trasferire informazioni fra entrambe le parti, programmare il PIC e anche aggiornare un programma senza doverlo togliere dal circuito dove si trova (quest'ultima operazione grazie a programmi residenti come Bootloader o Uploader che spiegheremo più avanti).

Enunciato

Si tratta di progettare un programma che legga lo stato logico di interruttori corrispondenti a RA4-RA0 e lo trasmetta via seriale a Hyper Terminal di Windows.

Nel caso in cui l'applicazione richieda comunicazione con un PC utilizzeremo il modulo USART nel suo modo di trasmissione asincrono.

Conoscendo la velocità a cui dobbiamo stabilire la comunicazione (9.600 **Baud**) possiamo configurare la trasmissione secondo la formula riportata nella figura.

Organigramma

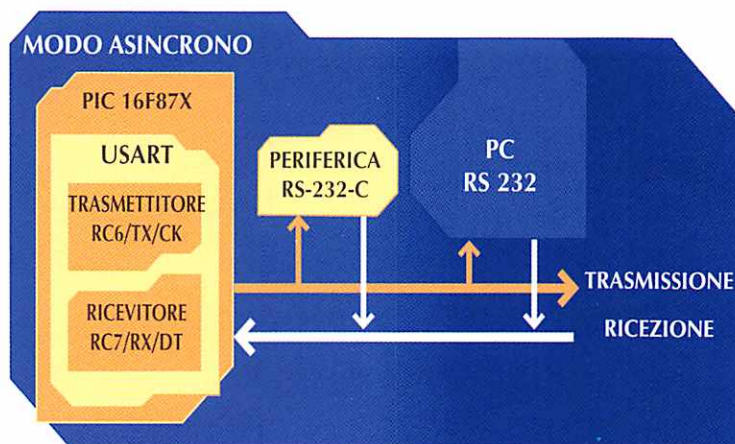
In un programma con queste caratteristiche conviene separare la parte del codice in subroutine, dato che esisteranno alcuni blocchi di codice che dovranno realizzare delle funzioni specifiche. Vi consigliamo di separare questi blocchi per svilupparli in modo indipendente.

Nell'organigramma della figura è rappresentata una soluzione generale all'esercizio, che ovviamente dovremo sviluppare con attenzione al momento di progettare il codice.

Codice

Intestazione e configurazione dei dispositivi

I commenti ci permetteranno di definire la funzionalità del programma, il PIC, la libreria delle definizione dei registri e grazie alle direttive ORG, la zona di memoria dove risiede il codice di programma.

Il programma principale inizia con l'etichetta Inizio configurando i dispositivi con cui vogliamo lavorare. La porta A sarà di ingressi digitali,

Schema di comunicazione del modulo USART modo asincrono.

$$Frec = \frac{Fosc}{k \cdot (x+1)}$$, *dove K vale* 16 o 64

Per calcolare il valore di x, sappiamo che la frequenza di comunicazione sarà Frec = 9600 Baud e che la frequenza di oscillazione del PIC è Foxc = $4 \cdot 10^6$ Hz

$$x = \left(\frac{Fosc}{k \cdot Frec} \right) - 1 = \left(\frac{4 \cdot 10^6}{16 \cdot 9600} \right) - 1 = 26 - 1 = 25$$

questo è il valore che verrà caricato sul registro SPBRG di configurazione.

Formule per configurare la comunicazione.

Organigramma dell'applicazione.

quindi dovremo configurare i registri TRISA e ADCON1. Nella porta C troviamo le linee di trasmissione RC6/Tx e RC7/Rx, essendo quest'ultima un ingresso configureremo il registro TRISC tenendo presente quest'ultima caratteristica. Configureremo il WDT mediante il registro OPTION_REG, infine determineremo la trasmissione. La prima cosa che dobbiamo fare è delimitare il tipo di trasmissione (modo asincrono ad alta velocità e da 8 bit) scrivendo il registro TXSTA, definire la velocità di trasmissione (9.600 **Baud**), caricando il valore ottenuto dalle formule sul registro SPBRG, e abilitare la trasmissione seriale impostando a 1 il bit SPEN del registro RCSTA.

Subroutine DATO di trasmissione dei caratteri

La trasmissione seriale si esegue byte a byte, quindi se vorremo inviare un testo lo dovremo fare inviando carattere per carattere. Dato che nella nostra applicazione vogliamo visualizzare su Hyper Terminal lo stato dei pin RA4:RA0 dovremo inviare come minimo 5 byte, uno per ogni carattere corrispondente allo stato di ogni pin di ingresso (carattere '0' o '1'). Sarà possibile anche migliorare la presentazione scrivendo un testo che preceda il dato in questione (ad esempio: porta A: xxxxxx) quindi sarà necessario trasmettere più byte dei minimi richiesti. Dato che dovremo ripetere il processo per ogni carattere da inviare, cioè molte volte, lo separeremo in una subroutine.

Ogni volta che entreremo nella subroutine dovremo resettare il flag o l'indicatore della trasmissione precedente, che viene attivato al termine di ogni trasmissione. Dovremo quindi passare il file da trasmettere al registro TXREG, e in seguito attendere che la trasmissione termini verificando il bit TRMT del registro TXSTA. Nella figura è possibile vedere il codice che corrisponde a questa subroutine.

Continuiamo con il programma principale

Abbiamo configurato i dispositivi e creato una subroutine di trasmissione, quindi ora dobbia-

```
;
;                              ESERCIZIO: LA USART
;                            ------------------------
;
;Il programma sarà un'applicazione del modo di trasmissione asincrono (RS-232).
;Programma che legge lo stato logico degli interruttori corrispondenti a RA4-RA0,
;trasmette questo stato logico via seriale al software hiperterminal di windows.


        LIST      P=16F870          ;Definiamo il nostro microcontroller
        INCLUDE   "P16F870.INC"     ;File dei registri interni

        ORG       0
        GOTO      INIZIO
        ORG       5
```

Intestazione del programma.

```
;Programma principale.

INIZIO          clrf    PORTC           ;Resettiamo la Porta C
                bsf     STATUS,RP0      ;Passiamo al banco 1
                movlw   b'00011111'
                movwf   TRISA
                movlw   b'00000110'
                movwf   ADCON1
                movlw   b'10111111'
                movwf   TRISC           ;RC6/Tx come uscita e RC7/Rx come ingresso
                movlw   b'11001111'
                movwf   OPTION_REG      ;WDT
                movlw   b'00100100'
                movwf   TXSTA           ;Attiviamo la trasmissione in modo
                                        ;asincrono a 8 bit ad alta velocità
                movlw   .25
                movwf   SPBRG           ;Carichiamo il registro generatore della velocità
                                        ;in baude in modo che si configuri con 9600
                bcf     STATUS,RP0      ;Torniamo al banco 0
                bsf     RCSTA,SPEN      ;Abilitiamo la porta seriale
```

Configurazione dei dispositivi.

```
;Trattamento della routine di trasmissione del dato.

DATO            bcf     PIR1,TXIF       ;Resettiamo il flag del trasmettitore
                movwf   TXREG           ;Scriviamo il dato
                bsf     STATUS,RP0      ;Passiamo al banco 1
CICLO1          btfss   TXSTA,TRMT      ;Verifichiamo se la trasmissione è terminata
                goto    CICLO1          ;No, non è ancora terminata
                bcf     STATUS,RP0      ;Torniamo al banco 0
                return
```

Inviamo un testo tramite la USART.

```
        LIST    P=16F870        ;Definiamo il nostro microcontroller
        INCLUDE "P16F870.INC"   ;File dei registri interni

aux1    equu    0x20
aux2    equ     0x21
aux3    equ     0x22

        ORG     0
        GOTO    INIZIO
        ORG     5
```

Definiamo tre variabili.

mo definire ciò che vogliamo trasmettere e come farlo.

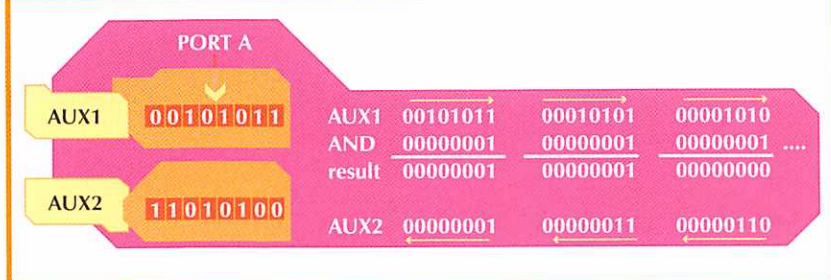
La prima cosa dao fare è inviare il testo "Porta A:" e come abbiamo appena visto lo dobbiamo fare carattere dopo carattere. Esistono una serie di comandi speciali che Hyper Terminal riconosce come ordini, quindi '/r' è un salto di linea (Return), '/t' è una tabulazione, ecc. Dobbiamo includere alcuni di questi comandi per migliorare la visualizzazione. Passeremo ogni carattere o comando da inviare sul registro di lavoro e chiameremo la subroutine di trasmissione, come si può vedere dal codice della figura. Acquisiremo lo stato degli ingressi e scriveremo il loro valore su una variabile ausiliaria. Dobbiamo trasferire i 5 bit meno significativi su RA4:RA0 uno per volta, iniziando da quello che corrisponderà a RA4 fino ad arrivare a RA0. Prima di mandarli sarà quindi necessario riordinare i bit.

Progettiamo ora una nuova subroutine che isoli il bit e lo porti riordinato su una nuova variabile ausiliaria. In questa subroutine entreremo cinque volte per ogni acquisizione della

```
                movlw   ':'
                call    DATO
                movlw   ' '
                call    DATO

                movfw   PORTA
                movwf   aux1

                clrf    aux2
                CALL    TRANSFER

                movlw   .4
                movwf   aux3
TRANS           rlf     aux2,f
                rrf     aux1,f
                CALL    TRANSFER
                decfsz  aux3,f
                goto    TRANS
```


Procedimento di riordinazione

```
;Routine per isolare il bit nel processo di riordino

TRANSFER        movfw   aux1
                andlw   1               ;Isolo il bit
                bcf     aux2,0          ;Imposto a 0
                iorwf   aux2,f          ;Sommo
                return
```

Codice per riordinare i bit.

```
                decfsz  aux3,f
                goto    TRANS

                movlw   .5
                movwf   aux3
RIPETE
                movfw   aux2
                call    BINTOASCII      ;Convertiamo il bit 0 in codice ASCII
                call    DATO            ;Inviamo il dato
                rrf     aux2,f          ;Bit successivo
                decfsz  aux3,f
                goto    RIPETE
                goto    CICLO

;Routine per convertire il valore del bit in ASCII

BINTOASCII      andlw   1               ;Isoliamo il bit 0
                addlw   '0'             ;Sommiamo il codice ASCII del carattere '0'
                return
```

Codice per l'invio del valore dell'ingresso.

porta, quindi creeremo una nuova variabile che funzioni da contatore (aux3). Definiamo queste variabili all'inizio del programma.

Per isolare i bit e riordinarli dobbiamo utilizzare le istruzioni di rotazione, in modo che il bit da isolare sia quello meno significativo, moltiplicare la variabile per b'00000001', passare il risultato alla variabile che contiene il valore ordinato sommandola a questa, e ruotare a sinistra il suo contenuto in modo che ripetendo l'operazione sia possibile alloggiare un nuovo bit nella posizione desiderata (vedere il grafico della figura). Tutto questo si traduce in codice nel modo indicato dalle immagini.

Dopo aver ordinato il valore acquisito lo dovremo trasferire bit a bit e a questo scopo ruoteremo il registro a destra isolando ogni bit. Il bit isolato avrà valore 0 o 1, ma dato che ciò che vogliamo trasferire è un carattere, ne cambieremo il valore in ASCII mediante una subroutine, in modo che possa essere inviato. Ruoteremo il registro 5 volte, dato che 5 sono i bit da trasmettere e ripeteremo il processo dalla trasmissione del testo iniziale.

Nella figura possiamo vedere il codice che corrisponde a quest'ultima parte del programma e la subroutine per convertire il bit in ASCII. Non dimenticate di terminare il programma con la direttiva END.

Il programma completo si trova sul secondo CD allegato all'opera con il nome di "ese15.asm".